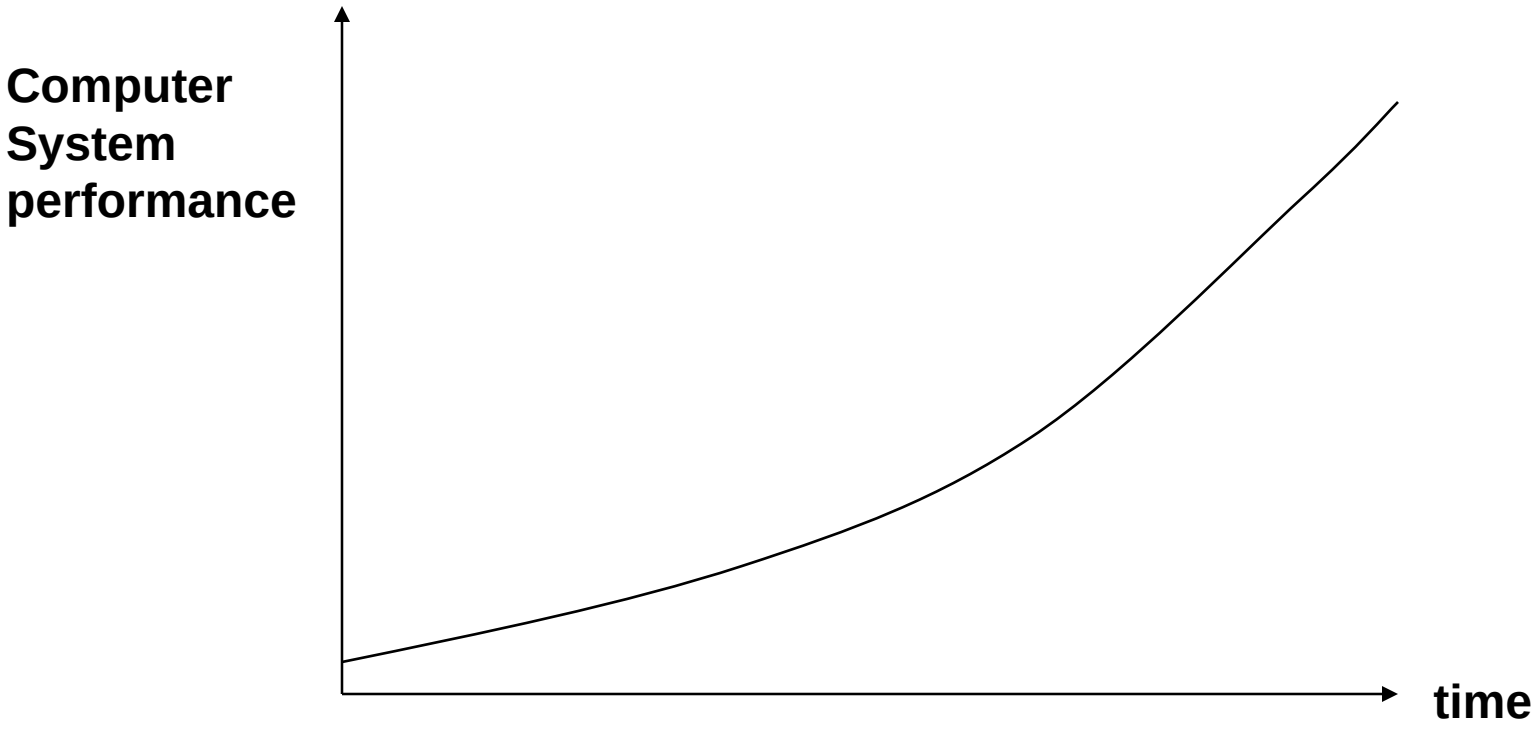
An Holistic Approach to Computer Architecture

Bob Colwell

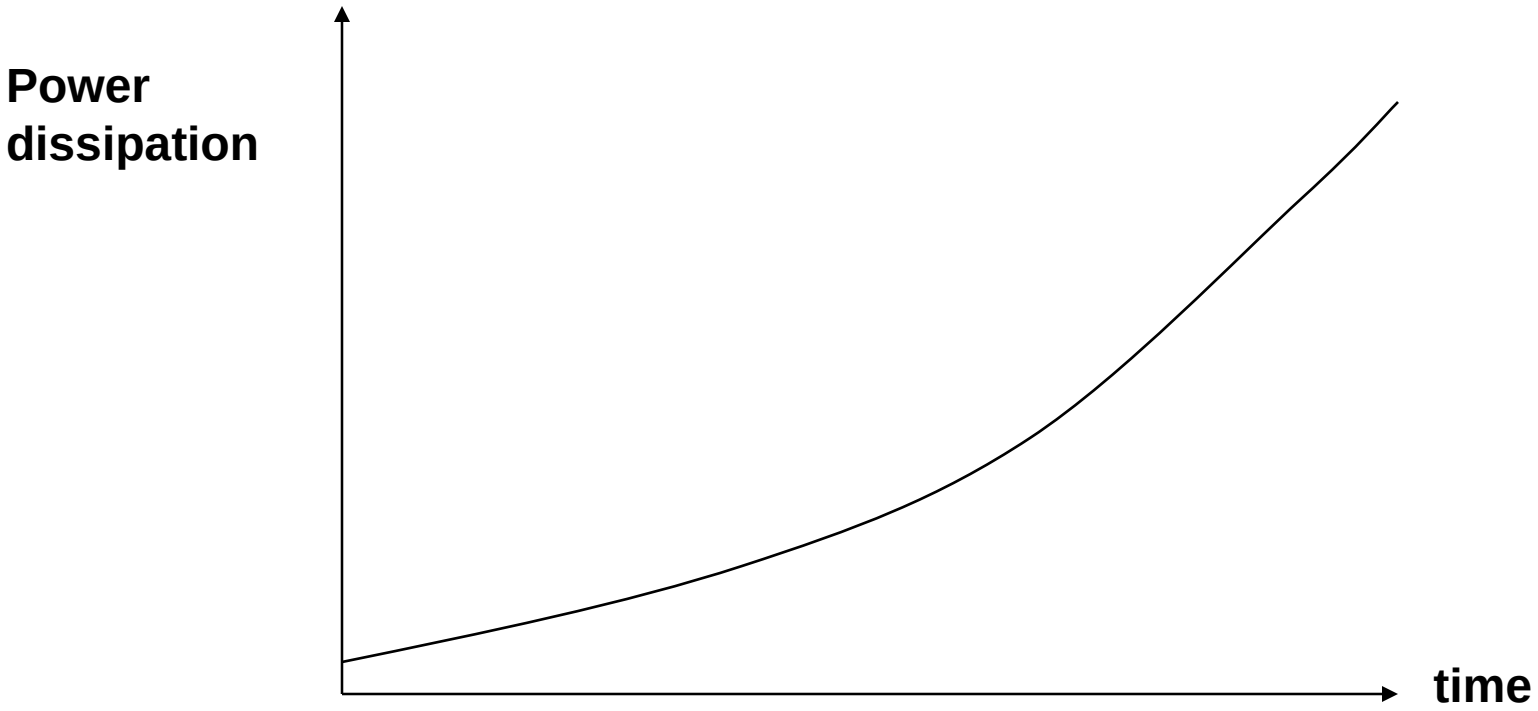
USC April 2004

This Curve Leads to $$$



And money is the ultimate arbiter of success in the microprocessor business

What Does This Curve Lead to?



Competing With Charcoal

Aesthetics Matter

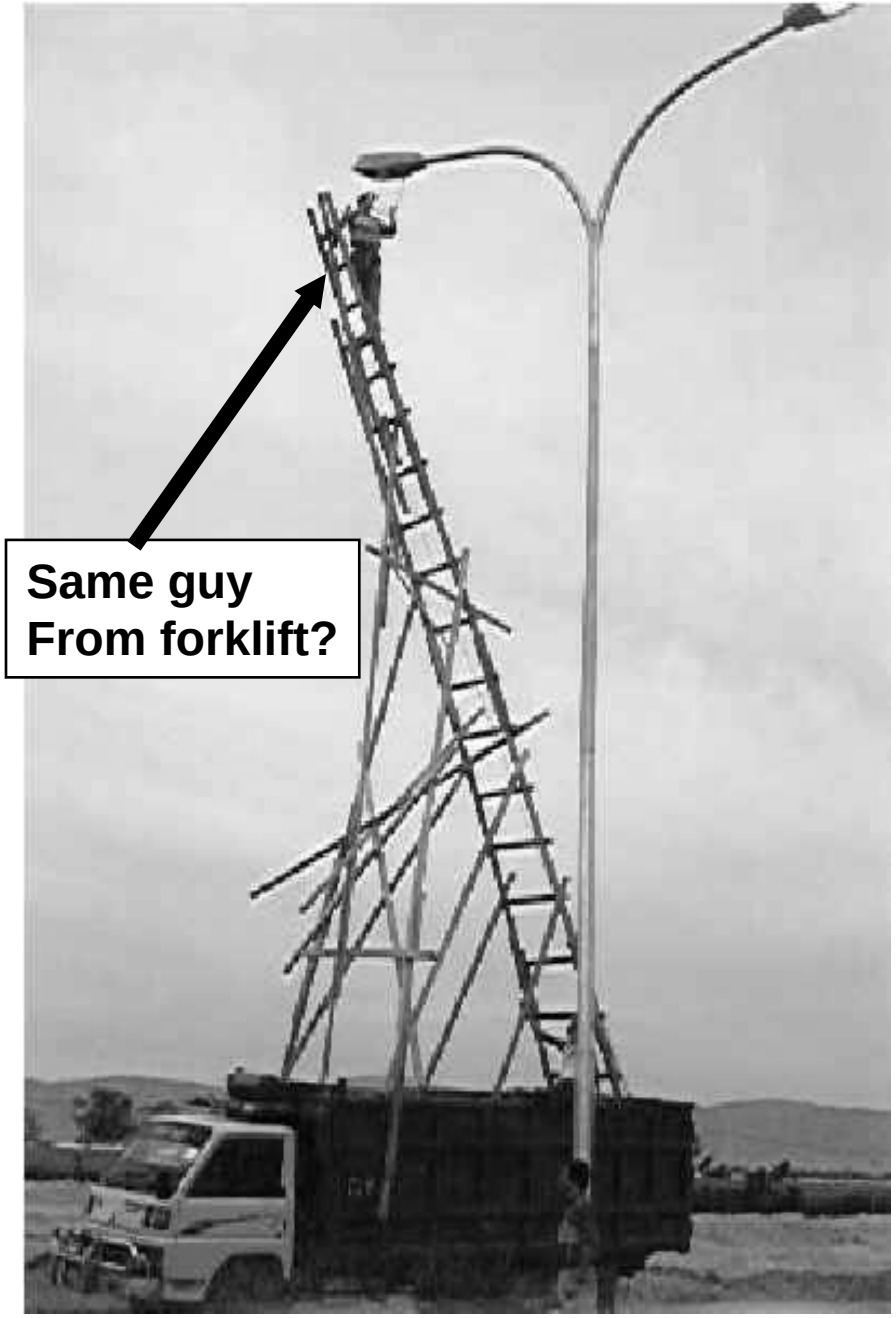
- **There are *always* problems**
- **Our job is to find *solutions***
  1. **There are elegant solutions**
  2. **There are workable solutions**
  3. **There are ugly hacks**


What is this guy thinking?
IA32+64-bit exts
IA32

Metaphors

- Secure software
- CPU's with no thermal protection
- WiFi WEP keys

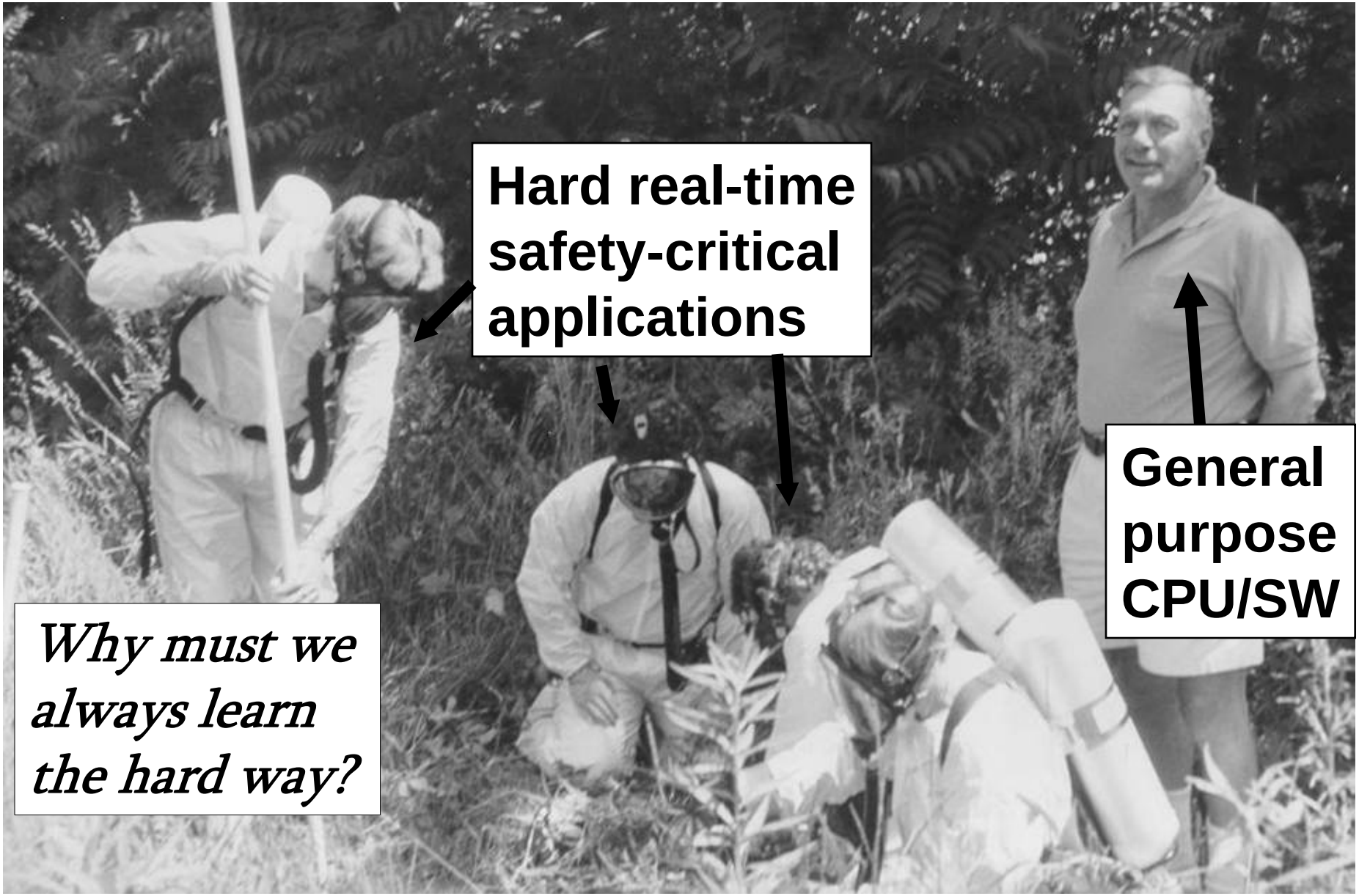
Using tried and true technology in new and dangerous ways



Metaphors

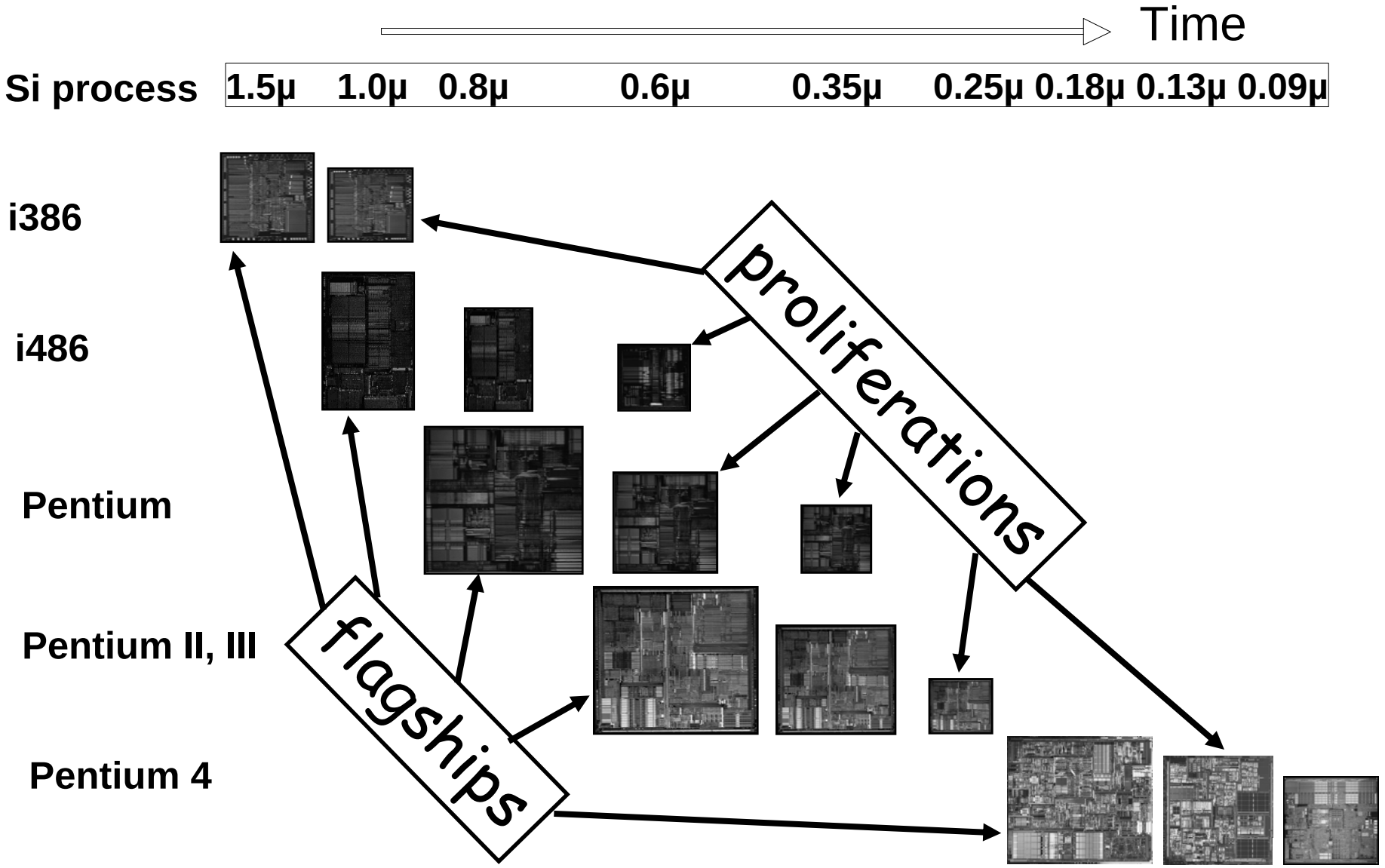
- Windows
- Complex CPUs
- SW bug patches

Complexity breeds fragility.
Fragility breeds surprises.
Surprises are never in your favor.

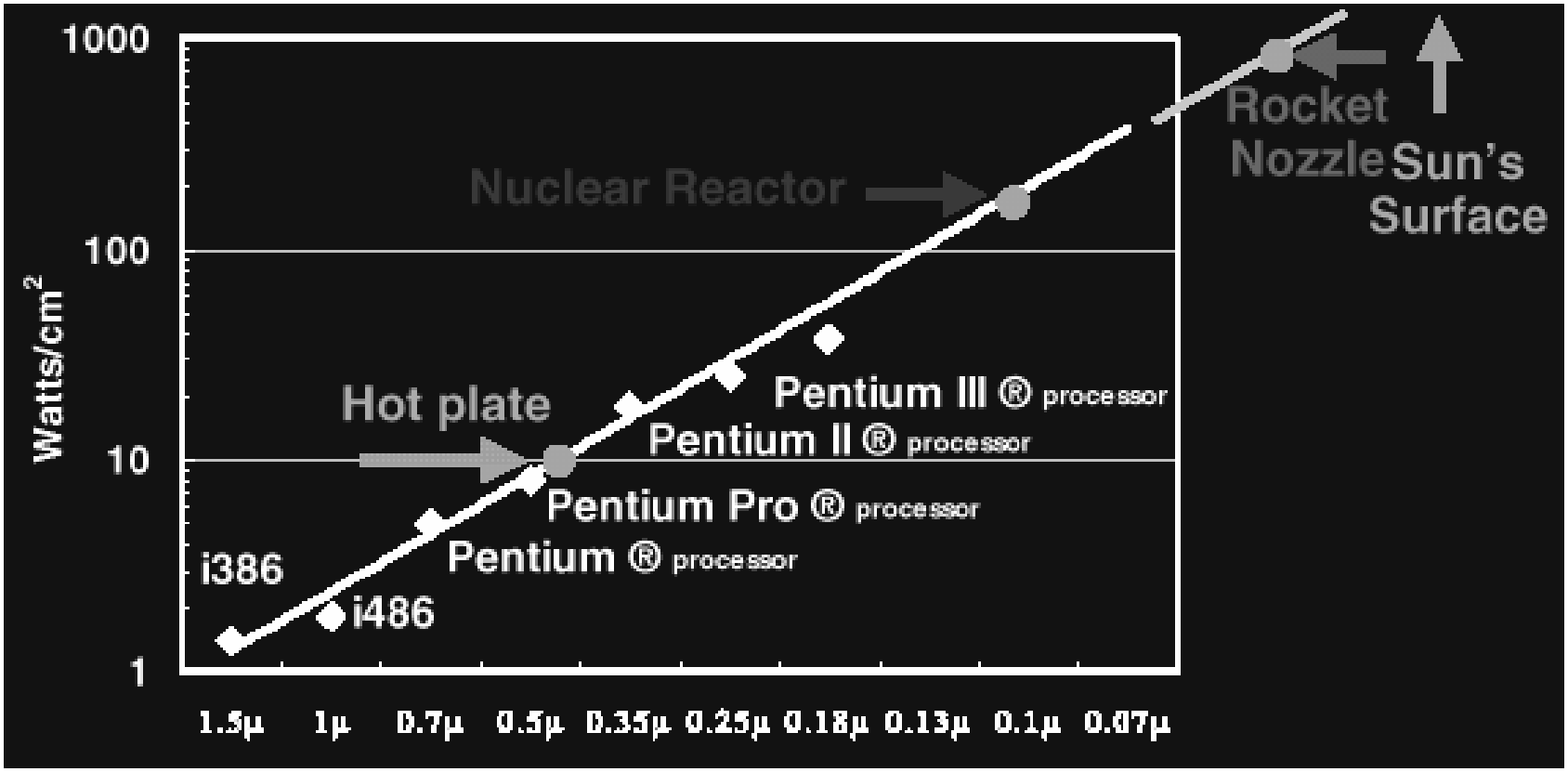


Hi guys. Hey, nice suits! Goin' diving or something?

The Magic of Moore's Law



Trouble Looms For Dr. Moore



- We're pushing perf & clk rates too hard

Moore's Law Is Not Free Any More

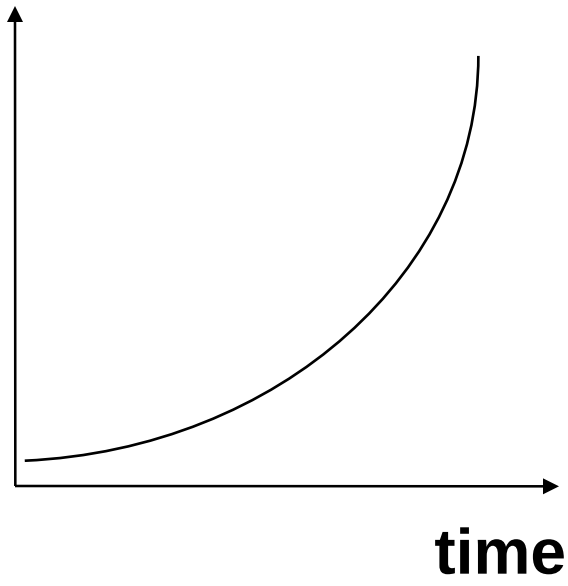
 # Transistors

Clock rate

Power

Bugs

Design $



- **Exponential trends are not sustainable**
- **<u>We have 3 here</u>**
- **Usable performance is leveling off**

- **But so *lucrative*!**

Tales of Two Architectures

- Itanium processor family and IA32
- Some lessons are common mode, some unique to one or the other
- And it isn't just technical…
  - We sell our high-tech products to the public
  - The public is weird
  - This does matter to CPU designers!

Itanium Processor Family Arch (IPF)

- **Pick your targets judiciously**
  - **In beginning, IPF was to replace IA32 by 1999**
    - **But Pentium 4 perf/clock projections killed that idea**
  - **Then it was to be workstations and servers**
    - **Hence heavy emphasis on floating point…but workstation market tanked**
  - **So then it was server-only**
    - **But mgement required IPF to be fastest at everything**
      - A great recipe for *mediocrity* at everything
  - **Consider: what ISA has ever survived as server-only?**
  - **What if the future is actually mobile**
    - **And you've rigged your ISA for server-space?**

IPF New Arch Gotchas

- There be dragons
  - Don't plan to sell your first chip
    - It won't be a product you're proud of
    - They won't buy it anyway, might as well benefit from that
      - Don't commit to ISA features that turn out duds (esp. not on basis of small but vocal set of early adopters)
  - Don't bend the benchmarks
    - Never base ISA perf projections on hand-coded inner loop
    - Especially if *compiler* is whole basis for architecture!
  - Evaluate features fairly
    - Predication, advanced loads, complex branching
    - Die size != reticle limit
    - Normalize economics (bus width, power dissipation)

IPF Planning Issues

- **Get your compatibility story straight**
  - **1/3rd of die for poor IA32 perf is…questionable**
  - **Could be binary translation is good enough**
    - **Or allow for dual-ISA implem in IPF in 1st place**
- **Pick your time horizon explicitly**
  - **25 years may be too long, but 10 years is too short**
  - **Forces you to follow Moore's law to numbers like 2KW CPUs, in face of increasing mobility of computing**
  - **Internet "erupted" < 10 yrs ago; Ability to anticipate future is quite limited…design accordingly**
- **Plan migration path explicitly**
  - **Chip roadmap must reflect long term plans for arch**

IA32: Pushing Elephants up Steep Hills

- Intel has had major product success pushing x86 into faster clocks & perf
- Downside of such tuning: <u>fragility</u>
  - Think thoroughbred racehorses: run fast as wind, but break legs easily
  - Very high end classical guitars have French polish and very thin tops

    Extremely delicate,but sound great

- Seems a universal law of nature
  - This applies to CPUs too

    Complexity++ => fragility++ ("glass jaws")

IA32: Fix One, Break Two

- **Partly function of design process**
  - **Takes 2-3 years to design high-end CPU**
  - **By time RTL mature enough for serious perf analysis, too late to make fundamental changes**
  - **So you tweak, and compound complexity**
  - **Easy to fix one perf divot by creating 2 more**
    - **And even this may be ok if new divots << old one**
    - **But complexity cost is permanent, affects all downstream efforts**

IA32: Universe IS Conspiring Against Us

- **Chaos at work? (Example 1)**
  - **P6 is self-scheduling engine with deep pipes**
  - **RS dispatch policy interacts with memory ordering buffer design, chipset & I/O to cause very long-lived patterns (100's of clock cycles)**
  - **Engine sometimes got into "slow patterns" that took 1000's of cycles to go away**
  - **A small change to design would have prevented it in case A but *caused* it in case B**

IA32: More Chaos (Example 2)

- Pentium 4 uses L1 cache data before tags have been checked
  - If mis-speculated, uop is yanked and redirected into "slow replay loop"
  - Replaying uop & dependents contend with new uops
  - Possible for entire machine to slow down for 100's or even 1000's of cycles until effect disappears

IA32: Even More Chaos

- **Industrial engineering using chaos theory**
  - **Applying chaos theory ideas to machine scheduling, improving throughput enormously by *inserting delays* when onset of slow system pattern detected**
  - **Not entirely surprising to CPU architects**
    - **Same effect in microcosm when two functional units of different latency share common bus**
    - **Often case that artfully inserting delays to the faster FU can speed up system overall**
- **Not well understood: how to detect onset of chaotic slowness, how to steer system into fast behavior patterns**

The Public Is Weird – $FDIV_1$

Chronology

1. Management said “make Pentium smaller”
2. Engineers searched design for opp’tys
3. Floating pt divider lookup table – shrink?
4. Formal proof: shrink ok
5. Validation assumed proof was correct
6. Bug found in P6 regression testing in OR
7. Prof Nicely found a few months later
8. New www amplified everything

The Public Is Weird – FDIV 2

- **Public … Intel.**
  - **… working …p, so give me one.**
  - **Nothing is bug-free, this bug doesn't … you.**
  - **I will judge … one it … You can't anyhow.**
  - **… your brand or suffer conseq.**
  - **Ok. We give up. Recall. $475M (ouch!).**

Irony 1: In the end it didn't save any die area.

Irony 2: it ended up being a net win for Intel's Pentium brand!

Execs finally "got it" that the rules had changed when it was pointed out that they would not take a new car home from the show room if there was a big scratch on the driver's door.

The Public Is Weird – $CPUID_1$

Chronology

1. **Marketing proposed productizing mfg id's**
2. **SW vendors had asked for onchip id's**
3. **We told them we couldn't do *unique* id's**
   - **So they renamed the feature "serial ID"**
   - **Being paranoid, we provided a way to disable**
4. **Feature debuted in Pentium III processor**
5. **Some people all over world went nuts**
6. **Intel turned feature off**

The Public Is Weird – CPUID$_2$

- **Public**
  - **I don't want you tracking me on the web.**
  - **We don't want anti-piracy.**
  - **Shut up. Turn this feature off or else.**
- **Intel**
  - **That's not what this feature is for. It's so we can identify Intel customers, and for anti-piracy.**
  - **Oh. You do know there are other visible id's in a PC?**
  - **Ok.**

Error 1: Intel took the business risk for the software vendors to get the potential return.
Error 2: Intel and everyone else in industry misjudged vehemence of vocal minority who don't trust anybody.

Common Arch Mistakes vol 1

1. **Know what is not compromisable**
   - **Perf, die size, schedule, power, features**
2. **Keep clear project goals**
3. **Avoid performance blindness**
   - **Goal-induced-myopia: EgyptAir 855, IPF, 3-mile-island**
4. **Don’t delegate big stuff: *you* track it**
   - **Die size, power, economics, perf, features, schedule**

Common Arch Mistakes vol 2

5. **Mice nibble away your margins: stop them**
6. **Quantify everything**
   - **And if you can't, know *why* you can't**
7. **Watch validation's travails**
   - **Early warning indicators of too much complexity**

Gross Generalities

- **Microprocessors may be outstripping general performance demand**
- **They *are* outstripping power & economics**

  Continuing to apply exponentially more resources to driving clock rates up by ever-more-complex and less-efficient microarchitectures, in an effort to provide higher performance that most people can't perceive, much less use, is not a rational plan

  It's just an old habit
- **Biz-as-usual will work until competition appears that adds real value**
  - **Mobile, wireless, security, human interface…fix *these***

Adjurations

- **Architects must take the long view**
  - **Nobody else will, not mgmt, not designers, not marketing**
- **Architect's job is to make valuable products**
  - **Not clever microarchitectures or instruction sets**
  - **Not "blue crystals" (caveat: clock rate)**
  - **Look for intersection between what *technology will be able to do* and *what buyers will want*, then sell that vision to rest of company**
  - **Pay careful attention to what implem technology affords**

    Leakage, noise, power, wireless, security

This ride isn't over, but the autopilot days are

Backup